by Liliia Makashova

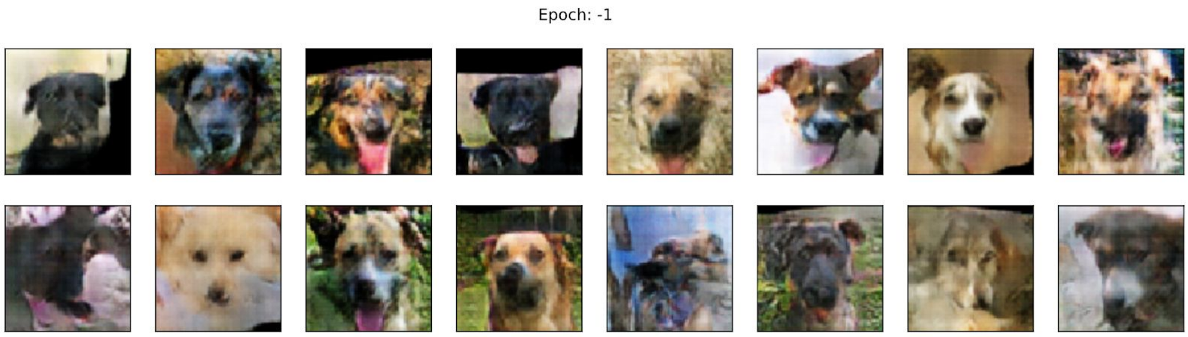
Project essay

Machine Learning course

LT2316

University of Gothenburg

GAN-DOGS : AN ATTEMPT TO RECREATE A VIRAL GAN-CATS PROJECT



Best results. Dog faces 64*64px at epoch 230

GANs.Introduction

Generative adversarial network (GAN) was first successfully implemented in 2014 by Ian Goodfellow and his team at the university of Montreal. The core idea of Goodfellow's method is the adversarial approach, e.g. to make two links in the network compete against each other, meaning that if someone wins, someone loses.

In the paper, the authors themselves explain the relationship between these two links - generator and discriminator - with a metaphor saying that the 'generative model can be thought of as analogous to a team of counterfeiters, trying to produce fake currency and use it without detection, while the discriminative model is analogous to the police, trying to detect the counterfeit currency' [4]. Therefore, such a competition makes both parts better - until something fake looks very real.

If speaking in more scientific terms, GAN proposed by Goodfellow uses indirect training through the discriminator and its weights are being updated dynamically at every step. Meanwhile, the generator tries to reproduce some data with the same distribution as in the training dataset. While the discriminator learns directly from the training data, the generator has access only to a so-called latent space of usually smaller dimension, and tries to generate new (fake) data from it [4].

The results of Goodfellow's project were not perfect, as their model after being trained on several different image datasets managed to produce images but a lot of noise in some cases made them incomprehensible. However GAN showed a lot of potential and spiked the research interest in this field[6].

Now GANs bring benefits to many industries such as design, medicine, graphics, science, gaming but also visual art, music etc.

Along with rapid development of GANs and their spreading throughout the industries, the concern of their possibly malicious usage has already been raised. Especially actively this discussion took place after the Style-GAN project was published. The purpose of the project was to generate high-resolution human faces and the project appeared to perform very well. After some resonance in the media, the use of AI generated human faces for social media and agitation in politics was banned. [3]

GAN-dogs project

Introduction

The GAN-dogs project was inspired by the rapid growth of interest for the Generative Adversarial Networks. More precisely, the GAN-cats project was a decisive point to attempt this task.

The GAN-cats toy example, that became viral on the Internet, performed extremely well with generating cat faces. Choosing dogs instead of cats for this project was a conscious decision, as in my opinion it brings more challenge. First of all, cats do not differ as much in sizes and 'faces' as dogs, however probably have the same variety in the range of their fur colourings [7, 8].

Datasets

The model from this project was applied for performance testing over two datasets.

The first choice of a dataset for the project was a Stanford Dog breeds dataset [1] . It's a pretty large image data consisting from over 22,000 annotated images of dogs belonging to 120 species. This dataset was collected for a task of fine-grained visual categorization. Working with this dataset assumes a more complicated task of generating images because dogs' pictures in this dataset may display dogs in various positions and frames (standing, lying, running and full-frame/only face pictures). Having such a variety of images, it was expected that it would inevitably influence the results of the networks in a worse way, as it would be difficult for the model to generalize having so many caveats to deal with.

Another dataset, which was submitted as a running example for this project was the DogFaceNet Dataset [5]. This dataset is more than two times smaller than the previous one, consisting of over 8000 images and 1393 classes (not breeds) of dogs. It was developed for dog face identification (recognition). The pure image size of this dataset is almost the same as in the GAN-cats project, which lets us compare the two projects more fairly.

The DogFaceNet dataset also brings additional perks to the Gan-Dogs project because of the dataset's structure. In the DogFaceNet dataset each class is represented by at least two images that are very similar (they show the same looking dog from different angles). In this case, class is not a breed, several classes can belong to one breed. They were divided into classes by colour. It was assumed that it will be fair to use the full dataset (several pictures per class) for the GAN-dogs project in terms of replicating the results of GAN-cats one, because, having bigger variety of angles from one class would let the algorithm catch more features of a dog of this breed.

As was mentioned before, dog image datasets have a bigger range of variance and probably cannot be compared to the cat ones in terms of 100% fairness. As of 2019, The International Cat Association recognizes 71 cat breeds [8], meanwhile there are over 400 dog breeds in the word[7]. Therefore, having a variety of cat breeds (e.g. cat images) being more than 3 times smaller than in dogs, this project was started with an assumption that reproducing the GAN-cats project's results will not go that smoothly.

GAN-dogs implementation

The implementation of the GAN dogs projects follows as close as possible the paper by Radford et al. from 2016 [6]. Since it was published 4 years ago, it might be considered outdated in the machine learning or artificial intelligence community, however, it is still able to produce pretty good results and be a good introduction step into GANs as a separate branch of AI.

For the author of this paper, GAN-dog project's main motivation is to be able to develop a model and make it somewhat work. The results' quality (e.g. realism of images) was considered a secondary goal because of the reasons described in the Datasets section.

Architecture

Radford's approach got a name currently known as DCGAN - Deep Convolutional Generative Adversarial Network. It had defined and reasoned sevelar unconventional steps for image generation tasks and scaled at that time an already known concept of Generative Adversarial Networks with Convolutional Neural Networks. Radford's model was the first successful implementation of such a simbiosys.

DCGAN's design has several defining features that make it recognizable and unique. First of all, a typical and very important step for CNNs in supervised learning is pooling layers. In the case of DCGAN it was removed and substituted by the strided convolutions, allowing the network to learn its own spatial downsampling [6]. As for a fully-connected linear layer, removing of which typically increased the quality of image classification models, in DCGAN it was kept both in the generator (as a first layer) and the discriminator (as the last one) [6].

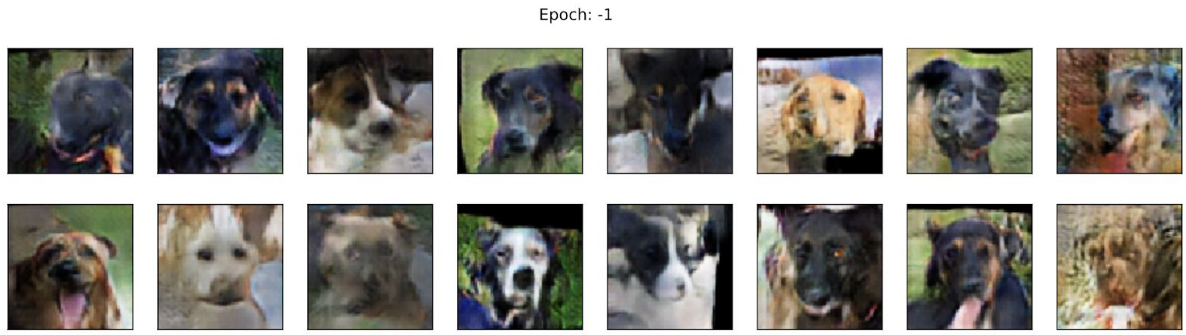
GANs' tendency to fail because of a generator collapse was fixed by applying batch normalization to all layers except the generator's output and discriminator's input layers. As for activation functions, the generator uses ReLU in all the layers, except the output, which uses Tanh; discriminator uses LeakyReLU in all layers with a slope of leak set to 0.2 [6].

Training GAN-dogs

Trainable hyperparamaters were set in accordance to those recommended in the paper [6]. Adam optimizer was set with a low learning rate of 0.0002 as well as with reduced momentum 'beta 1' to 0.5 instead of the default of 0.999. In the beginning, as in Radford's experiment the mini-batch's size was set to 128 and no data augmentation, except for resizing to a square of 128*128px (relatively high resolution) and center-cropping. The model was also trained with 64 mini-batch size and 64*64px images resolution to follow the GAN-cats project [2].

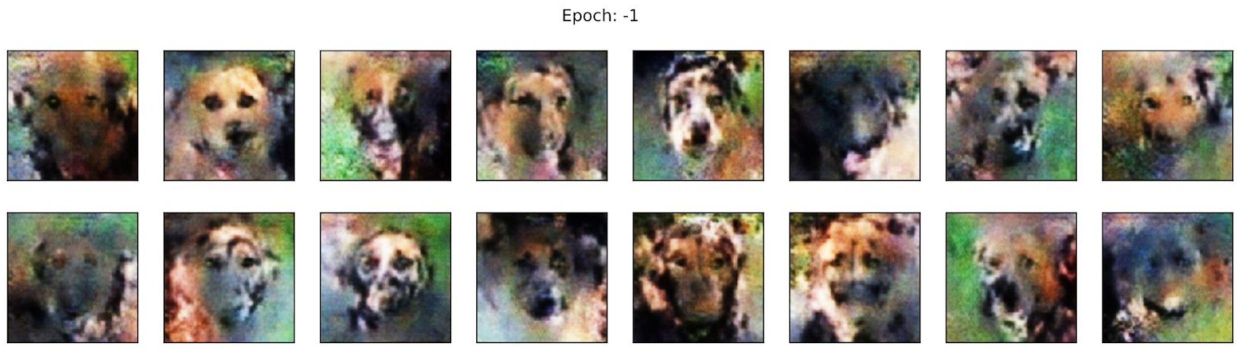
Evaluation

As there is a direct correlation between how fast models learn and tier generalization performance [6], after looking at the speed of going through a training loop we can say that the GAN-dogs model performs pretty well, managing to go through 10 epochs in 6 minutes while running over the DogFacesNet dataset with mini-batch set to 64 and with 64*64px image resolution.



Picture: images generated after 68 epoch. 64*64px resolution

Running the model over images with 128*128px resolution and batch size of 128, required changing some of the hyperparamaters and model's activations. The hyperparamaters for generation of higher resolution images included lower learning rate for both generator and discriminator set to 0.00005. The model's activations were changed, substituting LeakyReLU and BatchNorm with SELU to mimic the algorithm for generation of higher resolution images in the GAN-cats experiment [2]. This experiment showed expectedly slower learning (due to lower learning rate) and bigger loss for the generator over epochs. The image generation progress also appeared to be slower, however, the complete repetition of the experiment, which showed model convergence in the case with cat images after more than 20 hours of training, wasn't fully possible because of time constraints and resource limitations.



Picture: Dog faces high resolution 128*128px at epoch 230. Needs more training, but faces are visible.

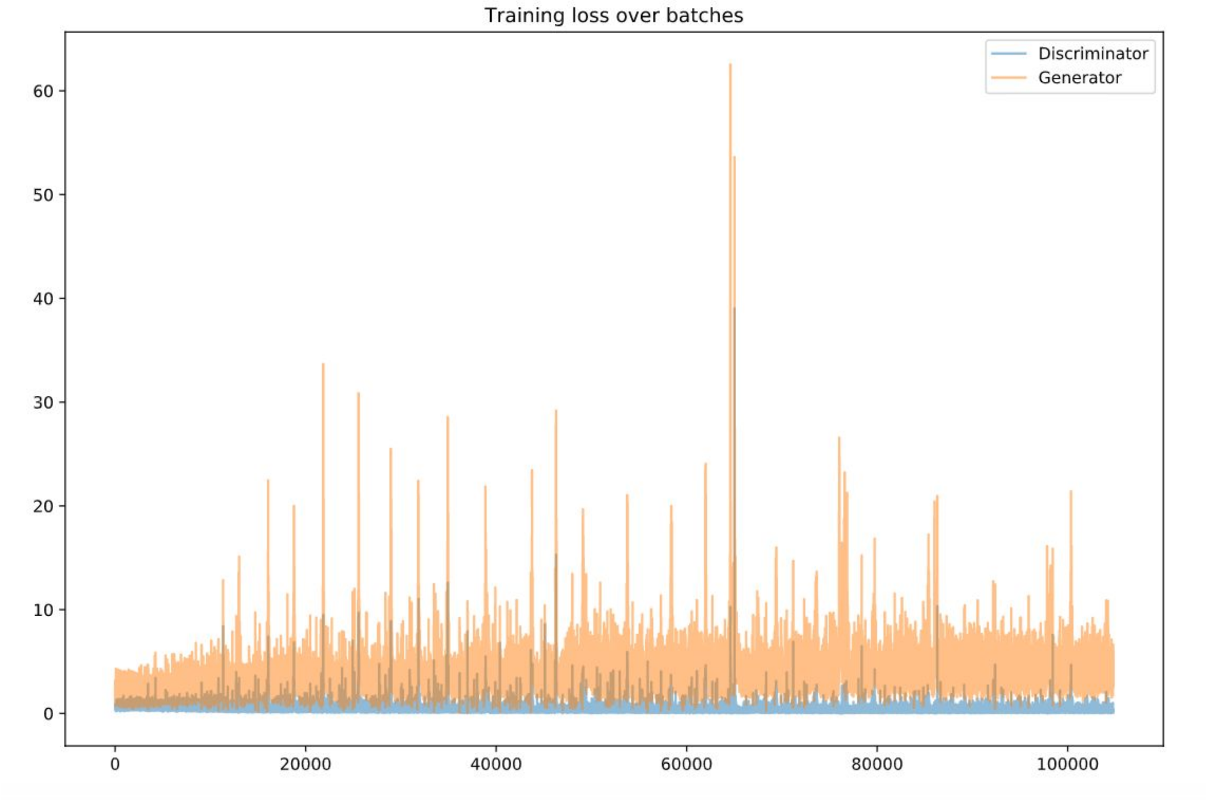
Running the same model over a bigger and more complicated StanfordDogs dataset shows two times slower performance and worse results if compared within the same range of epochs. It needs to be mentioned that to reach better performance over the StandfordDogs dataset, the model needs to be trained for even longer time and possibly more experiments should be conducted with the learning rate value as it’s the most important hypeparamater for DCGANs. In addition, to make this model with this dataset work, data preprocessing needs to be extended, possibly by defining a bounding box for a dog face (or figure) within the image and only then be given to the model as an input.

As the output quality was not a primary goal of this project, developing a better generated image from a StandfordDogs dataset was dropped after training a model over 100 epochs, due to time constraints. However, the confirmation of the model's ability to generate a more or less realistic image was gotten after comparing the outputs from the model over epochs - the image keeps getting better and more defined (e.g. more dog-like).

Generated images from the StanfordDogs dataset.

As for the performance of the model trained only on dogs' faces, it is able to generate pretty recognizable dog faces at 100 epochs, and continues to become less blurry when trained further. The training has rarely gone over 170 epochs due to time constraints and resource limitations.

The loss scores over both training processes (with DogFacesDataset and with StanfordDogs) look generally good (e.g clearly no overfitting), which proves that the hyperparamaters proposed by Radford are valid for this kind of task. Unlike classification problems in supervised learning, in DCGANs we should not expect a stable fall of loss scores over epochs. The losses graph typically looks like zig-zag as the discriminator and the generator keep 'fighting' against each other. If calculated on average loss should decrease, however, this was not shown on this graph (the graph shows the constant zig-zag on the more or less same level for discriminator and generator. The model has not yet converged but the discriminator and generator keep competing).



The evaluation method proposed by Radford et al., e.g. applying an algorithm as a feature extractor tool on supervised dataset and then evaluating the linear model fitted on top of those features [6], was not implemented for two reasons. The first one, being time constraints, is very self explanatory. The second one, lies in the goal of the project - make a working model and generate realistic images - which is a reconstruction of a toy GAN-cats project. The performance of the model in this case should be evaluated by simply assessing the generated image in terms of its 'dogness'.

Please note: two models, trained over the biggest allowed amounts of epochs were saved and were submitted with the assignment. The models saved are capable of generating dog faces in lower (64*64) and higher (128*128) resolution.

References

1. Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao and Li Fei-Fei. (2011) Novel dataset for Fine-Grained Image Categorization. First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
2. Alexia Jolicoeur-Martineau. (2017). Meow Generator. Retrieved from https://ajolicoeur.wordpress.com/cats/.
3. Doyle, Michael (May 16, 2019). "John Beasley lives on Saddlehorse Drive in Evansville. Or does he?". Courier and Press. Retrieved from https://www.courierpress.com/story/news/crime/2019/05/16/john-beasley-lives-saddlehorse-drive-evansville-does-he/3700111002/ .
4. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C. & Bengio, Y. (2014). Generative Adversarial Nets.. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence & K. Q. Weinberger (eds.), *NIPS* (p./pp. 2672-2680).
5. Guillaume Mouget. DogFaceNet Dataset1. (2019). Retrieved from https://github.com/GuillaumeMougeot/DogFaceNet/releases/tag/dataset.
6. Radford, A., Metz, L. & Chintala, S. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (cite arxiv:1511.06434 Comment: Under review as a conference paper at ICLR 2016).
7. Wikipedia. Dog type. Retrieved from https://en.wikipedia.org/wiki/Dog_type#Dog_types_and_modern_breeds.
8. Wikipedia. List of cat breeds. Retrieved from https://en.wikipedia.org/wiki/List_of_cat_breeds.